**Отчет по лабораторной работе № 2**

«Применение многослойной нейронной сети

для аппроксимации функций»

студента группы . Дата сдачи:\_\_\_\_\_\_\_\_\_\_

Ведущий преподаватель: оценка: подпись:\_\_\_\_\_\_\_

Вариант № 3

*Цель работы*: изучение математической модели многослойной нейронной сети и решение с её помощью задачи аппроксимации функций.

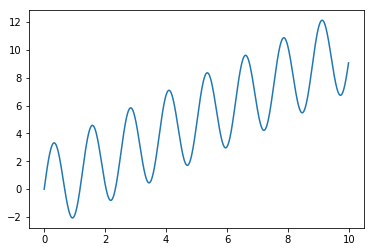
1. Подготовка данных

|  |  |  |  |
| --- | --- | --- | --- |
| Аппроксимируемая функция | Число  входов | Число выходов | Диапазон изменения аргументов |
| 3sin(5x)+x | 1 | 1 | [0, 10] |

Формирование обучающей, валидационной и тестовой выборок:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Обучающая | Валидационная | Тестовая | Всего |
| % | 60 | 30 | 10 | 100 |
| Объём выборки | 120 | 60 | 20 | 200 |

График аппроксимируемой функции:



Предобработка данных:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Метод | Параметры метода | Формула расчёта |
| Предобработка входов |  |  |  |
| Предобработка выходов |  |  |  |

1. Обучение и тестирование нейронной сети с одним скрытым слоем

Параметры архитектуры сети:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Число входов | Число выходов | Число нейронов в скрытом слое | Функция активации нейронов скрытого слоя | Функция активации выходного нейрона |
| 1 | 1 | 12 | sigmoid | Linear  *y* = *h* |

Схема нейронной сети:

|  |
| --- |
|  |

Параметры обучения:

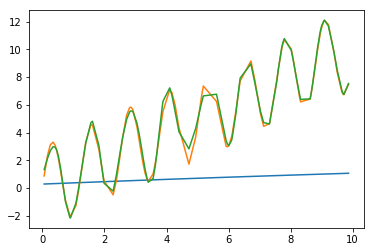
|  |  |  |  |
| --- | --- | --- | --- |
| Метод обучения | Скорость обучения α | Режим обучения | Функция потерь |
| GD | 0.01 | Mini-batch, bs = 20 | Quadratic loss |

Метод инициализации сети:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Критерий обучения: *E*(*w*) = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

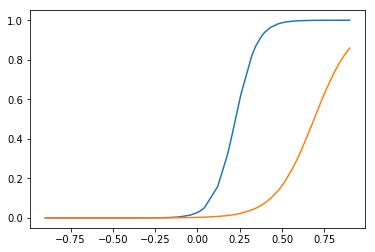
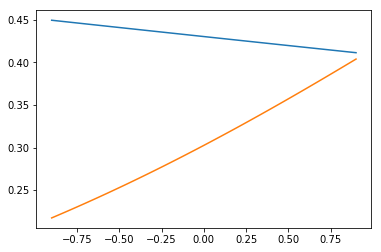
Критерий останова: выполнение 500000 эпох

Зависимость выхода *y*(*x*) сети от входа сети (изобразить три графика: до обучения, после обучения и график аппроксимируемой функции):

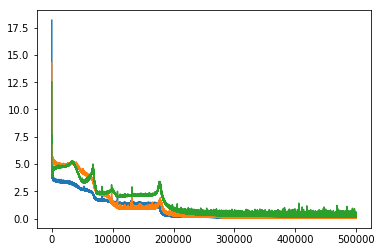


Зависимость выходов *yk*(*x*) нейронов скрытого слоя от входа сети (изобразить на одном графике):

*До обучения* *После обучения*



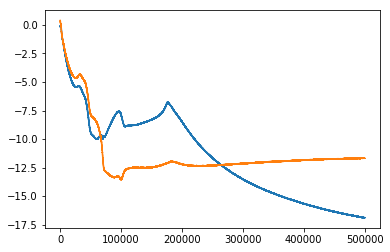
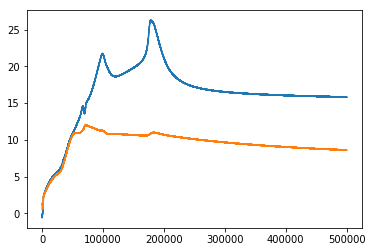
Зависимость ошибки сети *E*(τ) на обучающей, валидационной и тестовой выборках от времени обучения:



*Отметить на графике начало переобучения (если наблюдается)*

Зависимость синаптических коэффициентов сети *w*(τ) от времени обучения:

*Нейронов скрытого слоя* *Выходного нейрона*



Показатели качества обученной нейросетевой модели:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Обучающая | Валидационная | Тестовая |
| Макс. абс. ошибка | 1.2131 | 0.7154 | 0.7383 |
| С.к.о. ошибки | 0.0805 | 0.0720 | 0.1794 |
| RMSE | 0.2837 | 0.2683 | 0.4235 |

Обученная нейросетевая модель *обладает / не обладает* способностью к генерализации данных. Для улучшения качества аппроксимации требуется использовать *сеть с большим числом нейронов / сеть с меньшим числом нейронов / продолжить обучение имеющейся сети / изменить параметры метода обучения / изменить критерий останова / изменить режим обучения / обучить сеть заново из другой начальной точки.*

1. Улучшение качества аппроксимации

Параметры архитектуры сети:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Число входов | Число выходов | Число нейронов в скрытом слое | Функция активации нейронов скрытого слоя | Функция активации выходного нейрона |
| 1 | 1 | 12 | sigmoid | Linear  *y* = *h* |

Параметры обучения:

|  |  |  |  |
| --- | --- | --- | --- |
| Метод обучения | Скорость обучения α | Режим обучения | Функция потерь |
| GD | 0.01 | Mini-batch, bs = 20 | Quadratic loss |

Метод инициализации сети:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Критерий останова: выполнение еще 200000 эпох

Показатели качества обученной нейросетевой модели:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Обучающая | Валидационная | Тестовая |
| Макс. абс. ошибка | 0.9221 | 0.6472 | 0.6610 |
| С.к.о. ошибок | 0.0687 | 0.0662 | 0.1577 |
| RMSE | 0.2622 | 0.2574 | 0.3971 |

Выводы:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_